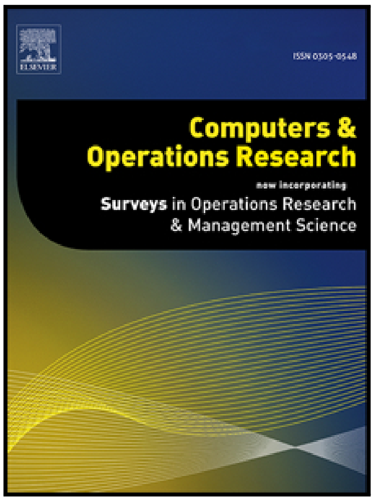
Accepted Manuscript




# Multiple domination models for placement of electric vehicle charging stations in road networks

Andrei Gagarin, Padraig Corcoran

PII: S0305-0548(18)30083-2
DOI: 10.1016/j.cor.2018.03.014
Reference: CAOR 4441

To appear in: *Computers and Operations Research*

Received date: 3 April 2017
Revised date: 26 March 2018
Accepted date: 28 March 2018

Please cite this article as: Andrei Gagarin, Padraig Corcoran, Multiple domination models for placement of electric vehicle charging stations in road networks, *Computers and Operations Research* (2018), doi: 10.1016/j.cor.2018.03.014

**Highlights**

- New approach to place charging stations for electric vehicles in road networks
- Computing reachability graphs for road networks and finding multiple dominating sets
- Can be used for placing refueling stations for alternative fuel vehicles in general
- Experiments with large-scale real world road networks of Boston and Dublin

# Multiple domination models for placement of electric vehicle charging stations in road networks

Andrei Gagarin[a,*], Padraig Corcoran[b]

[a]*School of Mathematics, Cardiff University, 21-23 Senghennydd Rd, Cardiff CF24 4AG, UK*

[b]*School of Computer Science & Informatics, Cardiff University, Queens Bldg, 5 The Parade, Cardiff CF24 3AA, UK*

**Abstract**

Electric and hybrid vehicles play an increasing role in road transport networks. Despite their advantages, they have a relatively limited cruising range in comparison to traditional diesel/petrol vehicles, and require significant battery charging time. We propose to model the facility location problem of the placement of charging stations in road networks as a multiple domination problem on reachability graphs. This model takes into consideration natural assumptions such as a threshold for remaining battery charge, and provides some minimal choice for a travel direction to recharge the battery. Experimental evaluation and simulations for the proposed facility location model are presented in the case of real road networks corresponding to the cities of Boston and Dublin.

*Keywords:* Road networks, Electric vehicles, Facility location problem, $k$-Domination, $\alpha$-Domination, Heuristic optimization

## 1. Introduction

Due to increasing concerns about the environment, the resulting policies and advances in technology, zero and low emission electric and hybrid vehicles are playing an ever more important role in road transportation. Despite the

*Corresponding author, phone: +44 (0)2920 688850

*Email addresses:* `gagarina@cardiff.ac.uk` (Andrei Gagarin), `corcoranp@cardiff.ac.uk` (Padraig Corcoran)

advantages of electric vehicles, their relatively limited cruising range (in comparison to traditional diesel/petrol vehicles) and significant battery charging time often provide major challenges to their usage.

As a result, in order for electric vehicles to be viable, it is necessary to have a sufficient number of charging stations which are appropriately distributed throughout a road network. Given a particular road network layout, determining appropriate locations and capacities for such charging stations is a challenging multi-objective optimisation problem with many constraints. One of the key objectives is to minimise the length of detours from a desired route which are necessary for recharging. On the other hand, constraints in this optimisation problem include requiring the number of charging stations to be reasonably small, ensuring the distance between consecutively used stations does not exceed the cruising range of electric vehicles, and that the capacities of the charging stations be sufficient enough to avoid bottlenecks. In this article, we focus on the problem of optimising the placement of charging stations such that the length of detours necessary for recharging is minimised subject to the constraint that the number of charging stations is reasonably small.

In the existing literature (e.g., see [15]), the problem of charging station placement is often modelled as a shortest path vertex cover problem for graphs. In this model, a vehicle is assumed to begin with a fully charged battery and follow a shortest path from an initial point to a final destination without much deviation. However, in many cases this assumption is not going to be valid, and the model in question is not going to be suitable. For example, mail or groceries delivery drivers are usually concerned with navigating in a way prescribed by delivery options (in time and space), and are not particularly concerned about shortest paths issues when navigating a certain area. Also, traffic jams, road closures and other temporary or sudden obstacles (e.g., a snow storm in Canada) may significantly influence the originally intended shortest path for driving. As a result, it is more natural and plausible to assume that drivers will become concerned about their remaining cruising range and battery charge only after the battery level falls below a certain low threshold, implying the remaining distance they can travel is quite limited.

In this work we propose a novel model for the placement of charging stations in road networks which is based on computing *multiple domination models* for a *reachability graph* corresponding to the original road network. A reachability graph models the set of locations which are reachable from a given location, where a location is reachable if its distance from the location in question is below a certain threshold. The reachability graph appropriately models the situation where a driver becomes concerned about their low battery charge and wishes to make a detour to a recharging station which is reachable from their current location. By considering multiple domination models on the reachability graph, we can compute a set of charging stations locations such that each location in the network can be served from several charging stations. That is, multiple charging stations are reachable from each such location. The driver therefore has several charging station options to select from and can in turn select the one that minimises the necessary detour.

In a practical context and possible applications, this approach can be used, for example, in the case of electric vehicles using the new emerging ultra-capacitor technology (e.g., see [18]). This kind of electric vehicles are known for a shorter cruising range, but much faster charging time. For example, in the case of new electric buses used for public transports in Minsk, Belarus, their driving range is currently about 20km, and the battery charging time is 5-8min (e.g., see [5]). Similar electric bus technology using ultracapacitors has been recently tested in Sofia, Bulgaria, with the battery charging time of 5-6min (see [9] and p.18 in [31]). Clearly, the cruising range should increase in the case of smaller size electric vehicles, and providing decreased battery capacity should respectively help to decrease the charging time.

Our approach can also be used to decide on efficient placement of fast charging and battery swapping stations. In this context, it can be used, for example, to decide on optimized development of charging infrastructure for urban taxi companies using electric vehicles, where only fast charging (about half an hour) makes sense during taxi service times (e.g., see [3]). Our proposed model is going to guarantee that a taxi driver, observing a low level of battery charge, will be able to reach a fast charging station (ideally at a taxi stand) and have a certain minimal choice of options for driving directions in his or her service area before serving another customer. Similarly, in the case of battery swapping scenarios (e.g., see [26]), a driver would have a guaranteed minimal choice of directions to change the battery at a reachable distance. The same approach can be used to decide on placing portable (mobile) fast charging and battery swapping stations as an extension of an existing network of permanent charging locations (e.g., see [21]). Finally, it can be adapted to decide on optimal locations of refuelling stations for alternative fuel vehicles in general (e.g., see [30, 21]).

The layout of this paper is as follows. In Section 2, we provide an overview of related work. The proposed facility location problem model is presented in Section 3. Section 4 describes the algorithms used to compute the reachability graphs and multiple domination models for the road network, plus provides some analysis and explains heuristic adjustments for the algorithms. An experimental evaluation using real road networks corresponding to the cities of Boston and Dublin is presented in Section 5. Section 6 provides a mixed-integer linear programming formulation for a capacitated generalisation of the problem, which is also used to obtain exact solutions for small-size problem instances to justify our heuristic benchmarks. Finally, in Section 7, we draw some conclusions and discuss possible future research directions.

## 2. Related work

There exist quite a large volume of recent literature related to electric vehicles and optimization in road networks focusing on different aspects of problem modelling and corresponding solution methods. For example, Poghosyan et al. [28] discuss possible scenarios of distribution of loads in the power grids and their dependence on temporal, spatial, and behavioural charging patterns for electric vehicles.

Given a set of charging stations and their locations fixed in the network, the authors in [29] propose a method for computing all locations which are reachable from a given initial location, assuming a specified number of battery recharges can be done. In this work, the locations of charging stations are assumed to be fixed, and there is no attempt to optimize the placement of charging stations in the network.

In [22], the authors consider a specific type of the general facility location problem called the electric vehicle charging station placement problem. In their work, they try to minimize construction costs for placement of charging stations in few pre-selected locations subject to a set of constraints. The problem is modelled using mixed-integer linear programming (MILP) with some non-linear constraints. The authors show that the problem is NP-hard and propose several solution methods by reduction to MILP problems and using heuristics. An experimental evaluation is first done with randomly generated small-size synthetic instances using MATLAB and generic MILP solvers. Then the model and methods are evaluated in the case of possible scenarios of building charging stations in Hong Kong by considering 18 pre-selected locations for potential construction of charging stations corresponding to different districts of the country. Notice that, in this model, the sites for potential construction of charging stations are pre-selected, and the average cruising distance of fully-charged electric vehicles is used to select the sites minimizing the total construction costs.

In [15], the authors model the problem of placement of charging stations as a "shortest path" cover problem in a graph of the road network $G = (V, A)$. One needs to find a smallest subset of vertices $L \subseteq V$ such that every minimal shortest path in $G$ that exceeds the electric vehicle battery capacity has a loading station placed in a vertex of the set $L$. The problem is then modelled as a special type of the Hitting Set problem: the collection of subsets of $V$ to be hit by the charging stations corresponds to the minimal shortest paths in $G$ that exceed the battery capacity. An adaptation of the standard greedy approach provides an $O(\log |V|)$-approximation algorithm to solve this problem. The instance construction and representation are described as the main challenges with respect to using limited computational memory and time resources. As a result, using different representations and searching for minimal shortest paths turns out to be a quite complicated task and is involved with many details. Overall, the problem does not seem to scale well, and the heuristic improvements for the implementation would be very challenging to reproduce.

A good description of optimization problems and different practical problem scenarios, mostly related to car-sharing systems employing electric vehicles, is presented in the overview paper [6]. Notice that models and optimization scenarios with construction of reachability graphs and finding multiple dominating sets in them, which are proposed and considered in our paper, roughly correspond to the strategic and tactical problems level described in [6]. A mathematical programming model that incorporates details of customer adoption behaviour and fleet management in car-sharing systems, including repositioning and charging electric vehicles under imbalanced travel plans, is considered in [20].

In [3], the authors estimate the potential charging demand by areas for a taxi company operating fossil-fuelled vehicles in the city of Vienna, Austria, and propose a method for placing in an optimal way a predefined fixed number of charging stations to maximize the coverage of the estimated charging demand. The problem is formulated as a MILP problem and solved by using a generic MILP solver (CPLEX). The authors point out that, in this case, only fast (Level 3) charging stations can be used at taxi stands to quickly recharge the taxis during their operational service time while waiting for the next customer. This is opposed to currently prevailing slow (Level 1) and standard (Level 2) charging of vehicles when they are not in use. Their solution is supposed to be further refined for each region, depending on taxi stands locations and other real-life constraints.

In [10], the authors use parking and personal trips information for a downtown area of Seattle, USA, to determine possible non-residential public parking locations for installing standard (Level 2) charging stations. In their behavioural models, they first predict (at least 15min) parking demand for different areas. Then, using different parking demand variables, the authors formulate a MILP problem to determine optimal locations (by areas) for placement of charging stations at parking lots. The MILP problem ensures that charging stations are not too clustered and have good accessibility by users. Some limitations of the model and optimization are discussed as well.

In [14], the authors present a study on possible efficient location of slow and standard (Levels 1 and 2) charging stations in public parking lots of a downtown area of Lisbon, Portugal. The area is characterized by a mixed high usage for both residential and workplace/business parking, with a low number of private parking spots. Therefore, many vehicles are parked for long time in public parking lots 24 hours per day. First, the authors estimate the recharging demand during the day and night time for smaller regions (census blocks) of the area by the numbers and different characteristics of households and the volume of employment and type of buildings. Then, a MILP formulation of the (maximal coverage) problem is presented to decide at which parking lots a limited pre-defined number of charging stations and with what number of supply points should be installed to maximize the total demand coverage. Four different scenarios are considered and discussed.

Given estimated demand for charging at specific locations, the authors in [8] propose three MILP problem formulations to decide on locations for slow and standard charging stations (for long-time parking). One is to maximize the satisfied demand coverage with respect to a fixed budget and taking into account the distance between the demand sites and actual charging station locations and the stations' capacities. The second MILP problem is an extension of the first one by allowing a transfer of the demand from one site to another in case the driver can charge at either of the two locations. Finally, the first two MILP models are refined to take into consideration variations of the demand for charging during different time intervals of the day. Simulations with the MILP models and optimization are described for the city of Coimbra, Protugal. To estimate the demand, the authors use data from a mobility survey for the city and represent the relevant demand areas by square grid cells and, in the case of high demand, by their subgrid cells (129 cells in total). The theoretical optimization results are compared to the actually implemented placement of nine charging stations in the city.

General MILP models for stochastic refuelling station location problems for fast-fill stations (Level 3 charging or battery swapping in the case of electric vehicles) are considered and described in [21]. The authors propose two-stage uncapacitated and capacitated MILP models for alternative-fuel vehicles, where the first stage decides on locations for permanent refuelling stations, and the second stage places portable (mobile) refuelling stations in a road network. Uncertain traffic flows depending on a number of time-dependent traffic scenarios in a road network are used as input parameters for the models. The resulting models are quite complicated, involved with details, and computationally intractable. Therefore, the authors propose heuristic methods to solve the problems and present simulations in the case of an intercity road network of the state of Arizona having 50 candidate facility nodes.

Different models, business scenarios and studies for development of a network of battery swapping facilities are presented in [26]. Agent-based simulations to show how different layouts of a limited number of refuelling stations for alternative fuel vehicles can influence adoption rates of the corresponding vehicles are presented in [30]. These simulations are based on randomly generated traffic flows for the city of Shanghai, China, and the alternative fuel stations layouts are optimized in different scenarios by using a genetic algorithm, with the main (largest) road network graph consisting of 532 nodes. A conceptual optimization model related to development of fast refuelling facilities in the case of medium- and long-distance travels by electric vehicles along a corridor is considered in [27].

## 3. Reachability graph and multiple domination models

For simplicity, we consider a road network represented by a weighed undirected simple graph $G^s = (V^s, E^s, w : E^s \to \mathbb{R})$, where the set of vertices $V^s$ corresponds to road intersections and dead-ends, while the set of edges $E^s$ corresponds to road segments connecting these vertices. The weight $w(e)$ on an edge $e \in E^s$ is the length of the corresponding road segment (in meters). An example of this graph model for the road network of the city of Boston is illustrated in Figure 1(a).

Given a road network graph $G^s = (V^s, E^s, w : E^s \to \mathbb{R})$, we define its *reachability graph* $G^r_t = (V^r, E^r_t)$ as a simple (unweighted) graph with $V^r = V^s$ and edges $uv \in E^r_t$ if and only if the length of shortest path (distance) between the corresponding vertices $u$ and $v$ in $G^s$ is less than a specified *reachability threshold* of $t$ km, i.e. $w(P^s_{uv}) \le t$, where $P^s_{uv}$ and $w(P^s_{uv})$ are a shortest path and corresponding distance between $u$ and $v$ in $G^s$, respectively. The reachability graph corresponding to the Boston road network of Figure 1(a) for $t = 3.0$km is illustrated in Figure 1(b). In this figure, red line segments are drawn between a given vertex and each of its neighboring vertices in $G^r_{3.0}$. The reachability graph $G^r_t$ appropriately models the situation where a driver becomes concerned about their low battery and wishes to make a detour to a recharging station which is reachable from their current location.

Notice that the reachability threshold $t$ to construct a reachability graph $G^r_t$ should normally satisfy the following lower bound derived from the road network graph $G^s$:

$$t \ge \max_{u \in V^s} \min_{v \in N(u)} w(uv), \tag{1}$$

where $N(u)$ is a set of all vertices adjacent to $u$ in $G^s$, and $uv \in E^s$. In other words, from any given point $u \in V(G^s)$, it should be possible to reach at least one of the neighbouring locations $N(u)$ using the remaining battery power (to eventually recharge the battery). This would imply the reachability graph $G^r_t$ has no isolates. Similarly, for better flexibility, more choice, and "safer" conditions for reaching possible recharging locations, one may impose the stronger lower bound for the threshold

$$t \ge \max_{e \in E^s} w(e). \tag{2}$$

This would mean it is possible to reach all the neighbouring locations $N(u)$ from any given point $u \in V(G^s)$ using the remaining battery power. The lower bound (2) would imply the vertex degrees of $G^r_t$ are at least the corresponding vertex degrees of $G^s$.

However, in the case of a small number of remote locations which are more difficult to reach in the network, it may be too demanding and expensive to satisfy the lower bound (2) or even (1) for the whole network. Therefore, when conditions of the lower bound (2) or (1) are not satisfied, the remote locations ("outliers" of the road network) should be treated separately. Thus, the "outliers" are considered in our models as well.

(a) (b)

Figure 1: (a) The road network for the city of Boston; (b) Neighbourhood of a vertex in the corresponding reachability graph.

Having constructed a road network graph $G^s$ and a corresponding reachability graph $G_t^r$, the problem of placing charging stations in the road network becomes a facility location problem which can be modelled on the graphs $G^s$ and $G_t^r$ as follows. In general, if $G$ is a graph of order $n$, then $V(G) = \{v_1, v_2, ..., v_n\}$ is the set of vertices of $G$, the degree of vertex $v_i$ is denoted by $d_i$ or $d(v_i)$, $i = 1, \ldots, n$, the minimum and maximum vertex degrees of $G$ are denoted by $\delta = \delta(G)$ and $\Delta = \Delta(G)$, respectively. The neighbourhood of a vertex $v$ in $G$ is denoted by $N(v)$. A subset $X \subseteq V(G)$ is called a *dominating set* of $G$ if every vertex not in $X$ is adjacent to at least one vertex in $X$. The minimum cardinality of a dominating set of $G$ is called the *domination number* of $G$ and denoted by $\gamma(G)$. Dominating sets in graphs are natural general models for facility location problems in networks.

Given an integer $k \geq 1$, a set $X \subseteq V(G)$ is called a *k-dominating set* of $G$ if every vertex $v \in V(G) \backslash X$ has at least $k$ neighbours in $X$. The minimum cardinality of a $k$-dominating set of $G$ is the *k-domination number* $\gamma_k(G)$. Clearly, $\gamma_1(G) = \gamma(G)$, and $\gamma_{k_1}(G) \leq \gamma_{k_2}(G)$ when $k_1 \leq k_2$. Given a real number $\alpha$, $0 < \alpha \leq 1$, a set $X \subseteq V(G)$ is called an *$\alpha$-dominating set* of $G$ if for every vertex $v \in V(G) \backslash X$, $|N(v) \cap X| \geq \alpha d_v$, i.e. $v$ has at least $\lceil \alpha d_v \rceil$ (i.e. $\alpha \times 100\%$) neighbours in $X$. The minimum cardinality of an $\alpha$-dominating set of $G$ is called the *$\alpha$-domination number* $\gamma_\alpha(G)$. It is easy to see that $\gamma(G) \leq \gamma_\alpha(G)$, and $\gamma_{\alpha_1}(G) \leq \gamma_{\alpha_2}(G)$ for $\alpha_1 < \alpha_2$. Also, $\gamma(G) = \gamma_\alpha(G)$ when $\alpha$ is sufficiently close to 0.

ACCEPTED MANUSCRIPT

The $k$- and $\alpha$-domination are two types of multiple domination in graphs. The concept of $\alpha$-domination differs from the $k$-domination in that a vertex must be dominated by a certain percentage ($\alpha \times 100\%$) of the vertices in its neighbourhood instead of a fixed number $k$ of its neighbours. Each of these two types of multiple domination can be used to model the situation when an electric vehicle driver starts to look for a conveniently located battery charging station and needs to have several options where to recharge the battery. In this paper, we focus on $k$-domination, which means that in any location (vertex) of the network (graph) the driver can use one out of $k$ possible options, $k = 1, 2, \ldots, \delta$. Clearly, in the case $k > \delta$, this model suggests that the vertices of degree less than $k$ are all included into the $k$-dominating set or ignored (i.e. treated separately). Therefore, without loss of generality, we can assume $k \leq \delta$.

The problems of finding exact values of $\gamma_k(G)$ and $\gamma_\alpha(G)$ and corresponding smallest size $k$-dominating and $\alpha$-dominating sets of vertices in graphs are known to be NP-complete [23, 12]. Therefore, it is important to have efficient heuristic algorithms and methods to find some reasonably small-size $k$- and $\alpha$-dominating sets in graphs. Also, it is important to have good theoretical bounds for $\gamma_k(G)$ and $\gamma_\alpha(G)$ to be able to estimate quality of a given solution set. The following two general upper bounds for the $k$- and $\alpha$-domination numbers have been obtained in [16, 17] by using a probabilistic method approach. These bounds generalize a classic upper bound for the domination number $\gamma(G)$. Also, the probabilistic constructions used in the proofs of these bounds allow us to design randomized algorithms to find $k$- and $\alpha$-dominating sets such that the expected cardinality of the set of vertices returned by the algorithm satisfies the corresponding upper bound.

Putting $\delta' = \delta - k + 1$ and $b_{k-1} = \binom{\delta}{k-1}$, where $0 \leq k - 1 \leq \delta$, we have:

**Theorem 1** ([17]). *For every graph $G$ with $\delta \geq k$,*

$$\gamma_k(G) \leq \left(1 - \frac{\delta'}{b_{k-1}^{1/\delta'}(1+\delta')^{1+1/\delta'}}\right) n.$$

For $0 < \alpha \leq 1$, we put $\widehat{\delta} = \lfloor \delta(1-\alpha) \rfloor + 1$ and $\widehat{d}_\alpha = \dfrac{1}{n}\sum_{i=1}^{n} \binom{d_i}{\lceil \alpha d_i \rceil - 1}$. Then we have:

**Theorem 2** ([16])**.** *For every graph $G$,*

$$\gamma_\alpha(G) \leq \left(1 - \frac{\widehat{\delta}}{\widehat{d}_\alpha^{1/\widehat{\delta}}(1+\widehat{\delta})^{1+1/\widehat{\delta}}}\right) n.$$

Clearly, given a reachability graph $G_t^r$, increasing the reachability threshold $t$ can only extend the neighbourhoods of vertices in $G_t^r$ to obtain $G_q^r$, $q > t$, i.e. $G_t^r$ is a spanning subgraph of $G_q^r$. Therefore, given a $k$-dominating set $X \subseteq V(G_t^r)$ in $G_t^r$, one can deduce some properties about this set $X \subseteq V(G_q^r)$ in the reachability graph $G_q^r$, where $q \geq t$. Clearly, having $k$ and the set $X$ fixed, $k \leq |X|$, and every vertex $v \in V(G_q^r)\backslash X$ is dominated by at least $k$ vertices in $X$. Then, as the reachability threshold $q$ increases, keeping the set $X$ fixed and considering $k$ as a parameter, the number $k$ can be eventually increased. When the reachability threshold $q$ is at least the diameter of the network graph $G^s$, the reachability graph $G_q^r$ becomes a complete graph, and every vertex $v \in V(G_q^r)\backslash X$ is dominated by all the vertices in $X$, so that we can set $k = |X|$.

If $q \leq t$, one cannot infer any domination properties of the $k$-dominating set $X$ of $G_t^r$ in the (spanning) reachability graph $G_q^r$. However, as $q$ approaches $t$, the set $X$ is going to start to behave like a $k$-dominating set with respect to the reachability graph $G_q^r$, and eventually the reachability threshold $t$ can be lowered. Some of the above properties are illustrated in the experimental results section of this paper.

## 4. Basic algorithms, heuristics, their implementation and complexity analysis

In this section, we describe the basic algorithmic ideas and routines to compute the reachability graphs and to find $k$-dominating sets in the reachability graphs. They are developed from and based on the theoretical results described in [1, 17] and our simulations and experiments with real road networks of Dublin and Boston. The $k$-dominating sets in the reachability graphs are facility location points for charging stations in the corresponding road network.

### *4.1. Computing the reachability graph*

The following procedure is used to compute the reachability graphs $G_t^r$. First, the vertices of $G^s$ are copied into $G_t^r$. Next, for each vertex $v$ in

$G_t^r$, we add an edge between $v$ and all the vertices in $G_t^r$ which are within the distance $t$ from $v$ in $G^s$. Here the distance between two vertices of $G^s$ is measured as the length of a shortest path. This is accomplished by performing a modification of the breadth-first search from the source vertex $v$ in $G^s$. Specifically, we employ Dijkstra's algorithm, but terminate the search when all vertices within a network distance $t$ of $v$ have been found. In our simulations, the graph $G^s$ is sparse. Therefore, to minimize running time, we implemented Dijkstra's algorithm using a binary heap based priority queue. This gives a running time of $O((|V^s| + |E^s|)\log|V^s|)$ for each call of this algorithm [4]. This algorithm is called for each $v \in V^s$ as the source vertex, giving a total running time of $O(|V^s|(|V^s|+|E^s|)\log|V^s|)$ for computing the reachability graph.

*4.2. Computing k-dominating sets in reachability graphs*

Algorithm 1 below is a randomized heuristic to compute a small-size (minimal by inclusion) $k$-dominating set in $G_t^r$ and is an adjustment of the corresponding randomized algorithm from [17]. It uses as an input a reachability graph $G_t^r$ and a positive integer $k$, $k \leq \delta(G_t^r)$. Algorithm 1 returns a (minimal by inclusion) $k$-dominating set $D$ in $G_t^r$, which provides a set of locations for charging stations in $G^s$ such that, from any given point (vertex) in $G^s$, a driver has at least $k$ different feasible options to reach a charging station when the remaining driving battery charge is enough for $t$ kilometers. The cardinality of the $k$-dominating set $D$ in $G_t^r$ returned by Algorithm 1 satisfies the upper bound of Theorem 1 with a positive probability, i.e. the expectation of the cardinality of $D$ satisfies the upper bound of Theorem 1.

The upper bound of Theorem 1 is known to be asymptotically best possible for general graphs on $n$ vertices in the case of 1-dominating sets (e.g., see [2]). In general, it is currently one of the best bounds for $\gamma_k(G)$ and likely to be asymptotically best possible for arbitrary $k$, $1 \leq k \leq \delta$. However, it turns out that the bound of Theorem 1 is not sharp enough in the case of particular reachability graphs of road networks for Boston and Dublin. As a result, randomized Algorithm 1 usually returns a non-minimal $k$-dominating set of an unreasonably large size. Therefore, instead of using the minimum vertex degree $\delta(G_t^r)$ of the reachability graphs $G_t^r$ to compute the probability $p$ and parameters $\delta'$ and $b_{k-1}$ in Algorithm 1, we use in our experiments the average vertex degree of $G_t^r$, i.e.

$$\bar{d}(G_t^r) = \frac{1}{n}\sum_{i=1}^{n} d_i.$$

In general, using $\bar{d}(G_t^r)$ instead of $\delta(G_t^r)$ in Algorithm 1 doesn't guarantee obtaining a $k$-dominating set satisfying the upper bound of Theorem 1. However, in the particular cases of road networks of Boston and Dublin, using $\bar{d}(G_t^r)$ in Algorithm 1 provides good computational results satisfying the upper bound of Theorem 1 as well. Notice that we have $k \leq \delta(G_t^r) \leq \bar{d}(G_t^r)$. Our implementation of randomized Algorithm 1 is enhanced with some other heuristics as well, and we run it several times to obtain smaller size $k$-dominating sets in $G_t^r$.

In the experiments, we have compared the results obtained by using the randomized approach of Algorithm 1 with those returned by a simple recursive greedy method described in Algorithm 2. Notice that, when $k = 1$, Algorithm 2 is a simple deterministic (greedy) approach derandomizing Algorithm 1 (e.g., see [1]). However, as already suggested by the results for $k = 2$ in [19], it can be a marvellous task to derandomize Algorithm 1 or similar randomized algorithms in general. The results returned by Algorithm 2 have been used as a benchmark to run Algorithm 1 several times to obtain better results (all satisfying the upper bound of Theorem 1).

The original $k$-dominating sets returned by Algorithms 1 and 2 are normally not minimal (by inclusion). Therefore, we have used a simple greedy procedure to reduce them to minimal $k$-dominating sets and to check that the final $k$-dominating sets are minimal. A pseudocode for this elimination of redundancy is presented in Algorithm 3. Notice that, in general, minimal by inclusion with respect to a property sets may have their cardinality significantly larger than the smallest-size sets (the latter must be minimal by definition). In other words, the cardinality of minimal $k$-dominating sets returned by Algorithms 1 and 2 may be larger than the $k$-domination number $\gamma_k(G)$ of the graph: as mentioned above, it is $NP$-hard to find $\gamma_k(G)$.

*4.3. Complexity analysis for the randomized algorithm*

Computing the binomial coefficient $b_{k-1}$ in Algorithm 1 is normally done by using the dynamic programming and Pascal's triangle, which has $O(\delta^2)$

**Algorithm 1**: Randomized $k$-dominating set

**Input**: A reachability graph $G_t^r$ and an integer $k$, $k \leq \delta$.
**Output**: A $k$-dominating set $D$ of $G_t^r$.

```
begin
    Compute the probability p = 1 − 1/ δ'√(b_{k−1}(1 + δ'));
    Initialize set A = ∅;                          /* Form a set A ⊆ V(G_t^r) */
    foreach vertex v ∈ V(G_t^r) do
        with the probability p , decide whether v ∈ A, otherwise v ∉ A;
        /* this forms a subset A ⊆ V(G_t^r)                              */;
    end
    Initialize B = ∅;
    foreach vertex v ∈ V(G_t^r)\A do
        if |N(v) ∩ A| < k then
            /* v is dominated by less than k vertices of A */
            add v into B; /* this forms a subset B ⊆ V(G_t^r)\A */;
        end
    end
    Put D = A ∪ B;                 /* D is a k-dominating set in G_t^r */
    If possible, remove some vertices from D to have a minimal
    k-dominating set D' in G_t^r;
    return D';
end
```

where $p = 1 - \dfrac{1}{\sqrt[\delta']{b_{k-1}(1+\delta')}}$.

**Algorithm 2**: Greedy $k$-dominating set

**Input**: A reachability graph $G_t^r$, an integer $k$, and $D \subseteq V(G_t^r)$.
**Output**: A $k$-dominating set $D$ of $G_t^r$.

```
while |{v ∈ V(G_t^r)\D : |N(v) ∩ D| < k}| > 0 do
    Set U = { v ∈ V(G_t^r)\D : |N(v) ∩ D| < k };
    Find u = argmax_{v∈V(G_t^r)\D} |N(v) ∩ U|;
    Put D = D ∪ {u};
end
```

**Algorithm 3**: Minimal $k$-dominating set

**Input**: A reachability graph $G_t^r$ and a $k$-dominating set $D$ of $G_t^r$.
**Output**: A minimal $k$-dominating set $D$ of $G_t^r$.
Order the vertices in $D$ as
$L = (v_1, \ldots, v_{|D|})\ :\ v_i \in D,\ |N(v_i) \setminus D| \leq |N(v_{i+1}) \setminus D|$;
**for** $i = 1$ *to* $n$ **do**
  **if** $D \setminus \{v_i\}$ *is* $k$*-dominating set of* $G_t^r$ **then**
    Put $D = D \setminus \{v_i\}$;
  **end**
**end**

time complexity in this case. The minimum vertex degree $\delta$ of $G_t^r$ can be computed in linear time in the number of edges $m$ of $G_t^r$. Notice that $O(\delta^2)$ does not exceed $O(m)$. Therefore, computing probability $p$ can be done in $O(m)$ time. It takes $O(n)$ time to find the set $A$, where $n$ is the number of vertices in $G_t^r$. The numbers $z = |N(v) \cap A|$ for each vertex $v \in V(G_t^r)\setminus A$ can be computed separately or when finding the set $A$. We need to keep track of them only while $z < k$. Since we may need to browse through all the neighbours of vertices in $A$, in total, it can take $O(m)$ steps to calculate all the necessary $z$'s for all $v \in V(G_t^r)\setminus A$. Then the set $B$ can be also found in $O(n)$ steps. Thus, in total, Algorithm 1 runs in $O(m + n)$ time. Since we heuristically use the average vertex degree $\bar{d}$ instead of the minimum vertex degree $\delta$ in our experiments with Algorithm 1, the complexity analysis of its implementation is slightly different, but can be easily derived from the analysis above.

It is possible to use simple heuristics when computing set $B$ in Algorithm 1. First, we can build set $B$ recursively, considering the undercovered vertices in $G_t^r\setminus A$ one by one. Then, we may want to include the most undercovered vertices, i.e. vertices $v \in G_t^r\setminus A$ with the smallest intersection $|N(v) \cap A|$, into $B$ first, and update set $A$ gradually by including the new vertices from $B$ directly into $A$ to form $A'$. This would recursively update the numbers $z = |N(v) \cap A'|$, make some of the undercovered vertices covered enough with at least $k$ neighbours in $A'$, and increase the coverage score $z = |N(v) \cap A'|$ for some $v \in G_t^r\setminus A'$ (in comparison to $|N(v) \cap A|$) to influence selection of the next vertex for $B$ in iteration. A heuristic "greedy extension" of $A$ procedure to find the sets $A'$ is similar to Algorithm 2.

Finally, since the initial recursively obtained $k$-dominating set $A'$ may be not minimal, we try to exclude some vertices from $A'$. This is implemented by removing vertices from $A'$ one by one and checking whether the $k$-domination property still holds. A heuristic procedure to guarantee the minimality (by inclusion) of the returned $k$-dominating set is described in Algorithm 3.

## 5. Experimental evaluation

In this section, in order to evaluate the proposed methodology, we describe our experiments with multiple domination models and corresponding algorithms in the case of two road network graphs $G^s$ corresponding to the cities of Boston in the USA and Dublin in Ireland. For the implementation, all the computer codes have been written using the programming language Python, and executable codes have been run on a laptop containing a 2.4 GHz Intel core i7-5500u processor, 16 GB 1600 MHz DDR3 RAM, and running Ubuntu 17.04.

### *5.1. Data*

The two road networks in question are illustrated in Figures 1(a) and 2(a), respectively, and are obtained from OpenStreetMap [11]. The graph $G^s$ corresponding to Boston consists of $21,542$ vertices and $31,112$ edges. It is contained within a rectangular region of width 15.5km and height 12.1km. The graph $G^s$ corresponding to Dublin consists of $55,162$ vertices and $64,437$ edges. It is contained within a rectangular region of width 29.5km and height 24.6km. Notice that both road network graphs are either planar or "almost" planar: when considering them embedded in the plane as road maps, the edge crossings are only possible in the case of road bridges and tunnels. Moreover, these two graphs are sparse in terms of the number of edges $m$, which satisfies the linear upper bound in terms of the number of vertices for planar graphs, $m \le 3n - 6$, $n = |V(G^s)|$, as opposite to the general worst case quadratic upper bound $m \le \frac{n(n-1)}{2}$, i.e. $m = O(n^2)$.

The most appropriate reachability threshold $t$ for the reachability graph $G^r_t$ is a function of a large number of parameters. This includes the number of electrical vehicles which require charging, the number of charging stations one is able to install, the number of charging options one wishes to offer, and the cost of installing a charging station. Determining this threshold would probably best be done by consultation with city planners. In this paper,

(a)

(b)

Figure 2: (a) The road network for the city of Dublin; (b) Neighbourhood of a vertex in the corresponding reachability graph.

we assume the most appropriate reachability threshold for both cities' road networks and electrical vehicles is 3.0km.

For each road network graph $G^s$, we computed the corresponding reachability graph $G^r_{3.0}$. These graphs are illustrated in Figures 1(b) and 2(b). The reachability graph $G^r_{3.0}$ corresponding to Boston contains $21,542$ vertices and $23,052,466$ edges (approx. 9.94%). The reachability graph $G^r_{3.0}$ corresponding to Dublin contains $55,162$ vertices and $54,306,700$ edges (approx. 3.57%). The CPU time required to compute these reachability graphs was 464 and $1,054$ minutes, respectively. All $k$-dominating sets are computed using these two reachability graphs.

### 5.2. Computing $k$-dominating sets

For each of the reachability graphs $G^r_{3.0}$, one $k$-dominating set was computed using the greedy algorithm, and ten $k$-dominating sets were computed using the randomized algorithm for $k = 1, 2, 4$. Table 1 displays the cardinalities of the $k$-dominating sets computed using the greedy algorithm and the cardinalities of the smallest $k$-dominating sets computed using the randomized algorithm for each of the cities and each value of $k = 1, 2, 4$. In four out of the six cases, the randomized algorithm computed a smaller dominating set than the same multiplicity dominating set returned by the greedy algorithm. The two 2-dominating sets for the city of Boston are displayed in

| Network | Boston | | | | Dublin | | | |
|---|---|---|---|---|---|---|---|---|
| Algorithm | Greedy | Randomized | | | Greedy | Randomized | | |
| | | min | mean | std | | min | mean | std |
| $k = 1$ | 32 | 31 | 33.2 | 1.6 | 110 | 111 | 114.4 | 2.0 |
| $k = 2$ | 64 | 56 | 61.2 | 2.8 | 214 | 215 | 220.7 | 3.7 |
| $k = 4$ | 122 | 115 | 120.3 | 3.2 | 413 | 411 | 418.0 | 3.4 |

Table 1: Cardinalities of the $k$-dominating sets computed using the greedy algorithm and the smallest (average, standard deviation) size $k$-dominating sets computed using the randomized algorithm for each city and each value of $k$.

| Network | Boston | | | Dublin | | |
|---|---|---|---|---|---|---|
| Algorithm | Greedy | Randomized | | Greedy | Randomized | |
| | | mean | std | | mean | std |
| $k = 1$ | 24 | 53 | 5 | 87 | 145 | 10 |
| $k = 2$ | 33 | 89 | 7 | 121 | 275 | 14 |
| $k = 4$ | 55 | 107 | 7 | 192 | 516 | 18 |

Table 2: CPU time of the greedy algorithm, and mean and standard deviation of the CPU time of the randomised algorithm to find $k$-dominating sets (in minutes).

Figure 3. The two 4-dominating sets for the city of Dublin are displayed in Figure 4.

A visual inspection of Figures 3, 4, and others reveals that spatial locations of the elements in the dominating sets tend to be more spatially clustered when computed using the greedy algorithm. This can be attributed to the greedy nature of the approach: vertices of high degree in the corresponding reachability graphs tend to be spatially clustered, and the greedy algorithm will add these high degree vertices to the dominating set first. On the other hand, the randomized algorithm initially adds a random set of vertices to the future dominating set, and these vertices are likely to be spatially distributed in a more uniform way.

The CPU time required by each algorithm to compute the $k$-dominating sets in the reachability graphs for the cities of Boston and Dublin are reported in Table 2. For the randomized algorithm, the mean and standard deviation of the ten run times are reported instead of showing them individually. An interesting point is that the run time for the randomized algorithm is generally greater than that of the greedy algorithm, which can be attributed to a more subtle and sophisticated nature of the randomized approach.

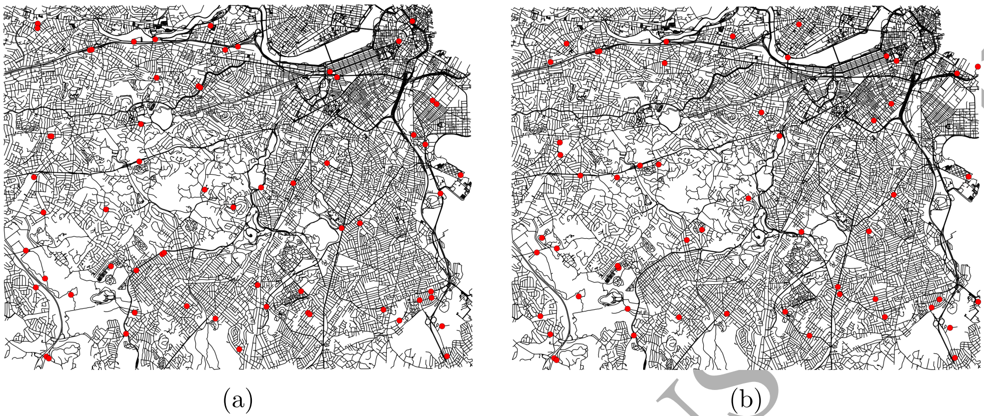


Figure 3: (a) The 2-dominating set of 64 vertices computed by the greedy algorithm and (b) the smallest 2-dominating set of 56 vertices computed by the randomized algorithm (both for the city of Boston; the vertices in the 2-dominating sets are in red; see Table 1).

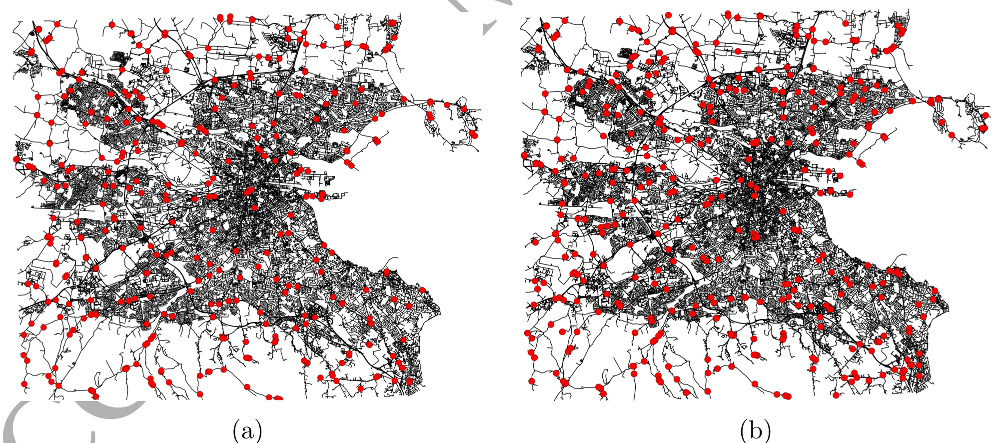


Figure 4: (a) The 4-dominating set of 413 vertices computed by the greedy algorithm and (b) the smallest 4-dominating set of 411 vertices computed by the randomized algorithm (both for the city of Dublin; red dots indicate the vertices in the 4-dominating sets; see Table 1).

*5.3. Reachability of stations*

Given a fixed $k$-dominating set $X$ in a reachability graph $G_t^r$ corresponding to a road network graph $G^s$, the number of elements in $X$ reachable from a given vertex in $G^s$ is a non-decreasing function of distance. To examine this phenomenon, we consider the smallest 2-dominating sets computed using the randomized algorithm for Boston and Dublin. These two sets contain 56 and 215 elements, respectively. The set corresponding to Boston is illustrated in Figure 3(b).

We computed the mean and standard deviation of the number of vertices in $X$ reachable from a vertex in $V(G^s)\setminus X$ as a function of distance. These values are displayed in Table 3. An analysis of this table reveals the following facts. Despite the fact that the dominating sets were computed for a reachability graph with the reachability threshold of 3km, the mean number of vertices in $X$ within a distance of 1km of a vertex in $V(G^s)\setminus X$ for each of the cities is 0.5. Furthermore, for both cities, the mean number of elements in $X$ within the distance of 3km from a vertex in $V(G^s)\setminus X$ is significantly larger than 2. This means more support and flexibility than only 2 a priory guaranteed options for recharging electrical vehicles in many points of these road networks.

On the other hand, increasing the reachability threshold to some $q > t =$ 3km and keeping the set of vertices $X$ fixed in $G_q^r$ should allow us to increase the minimum multiplicity of coverage of each vertex $v \in V(G^s)\setminus X$ by the vertices in $X$ to have $X$ as a $k$-dominating set with $k > 2$ in $G_q^r$. We have computed the minimum multiplicity of coverage by the same 2-dominating sets $X$ in the corresponding reachability graphs $G_q^r$ with the reachability threshold $q$ increasing to 4, 5, and 6km. As shown in the corresponding columns of Table 3, this increase of the reachability threshold haven't allowed us to increase the minimum number of options for the city of Boston, but have turned the 2-dominating set $X$ of $G_{3.0}^r$ into a 3-dominating set in $G_{6.0}^r$ for the city of Dublin. In other words, the drivers in Dublin are going to have at least 3 options available within the distance of 6km for recharging the batteries when using the same 2-dominating set $X$ from $G_{3.0}^r$.

*5.4. Detour required*

The number of options available for recharging electrical vehicles in a road network $G^s$ increases as a function of the multiplicity value $k$ in the corresponding $k$-dominating set. In turn, this may reduce the length of detours required for recharging electrical vehicles. To quantify this phenomenon, we

| Network | Boston | | | Dublin | | |
|---|---|---|---|---|---|---|
| Stats | Mean | Std | Min | Mean | Std | Min |
| 1 km | 0.5 | 0.7 | 0 | 0.5 | 0.7 | 0 |
| 2 km | 1.9 | 1.1 | 0 | 2.0 | 1.1 | 0 |
| 3 km | 4.3 | 1.4 | 2 | 4.6 | 1.5 | 2 |
| 4 km | 7.3 | 1.9 | 2 | 8.5 | 2.3 | 2 |
| 5 km | 10.9 | 2.6 | 2 | 13.7 | 3.1 | 2 |
| 6 km | 14.9 | 3.7 | 2 | 20.0 | 4.2 | 3 |

Table 3: Three statistics for the number of charging stations reachable from a vertex outside of the charging station locations for the road networks of Boston and Dublin computed as a function of distance.

consider the situation where a driver of an electrical vehicle wishes to travel from a source location to a destination, but first needs to have their vehicle recharged. Therefore, the driver considers all charging stations within the distance of 3km from the source and charges their vehicle at a charging station which minimizes the detour. Here the detour is the difference between the distance from source to destination and the sum of distances from source to the charging station and from the charging station to destination.

To illustrate this, consider Figure 5 and the situation where the source and destination are represented by red dots in the left and upper right of the figure, respectively. Considering the smallest 2-dominating set computed using the randomized algorithm (56 vertices, see Table 1 and Figure 3(b)), there are five charging stations within the distance of 3km from the source. These five charging stations are represented by green dots in Figure 5. The route which minimizes the detour is represented by the blue line in the figure, and the detour in question is only 92 meters.

For each of the cities of Boston and Dublin, we have selected two hundred random pairs of source and destination locations, and for each pair of the locations, the corresponding detour for recharging was calculated. Considering the smallest $k$-dominating sets computed using the randomized algorithm for different values of $k = 1, 2, 4$, the corresponding mean and standard deviation of detours required for recharging are displayed in Table 4. As expected, for both cities, the mean and standard deviation values decrease as the multiplicity of domination parameter $k$ increases.

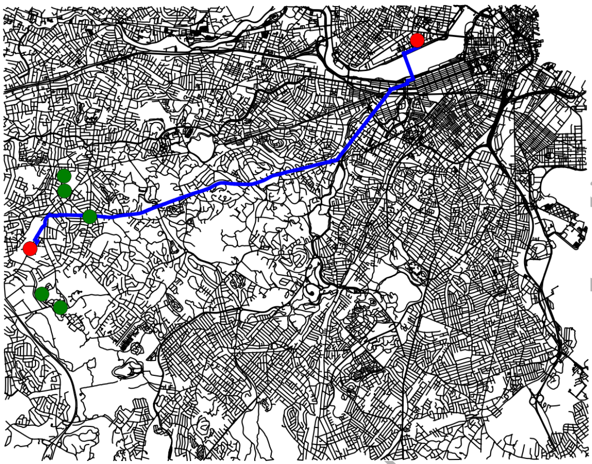


Figure 5: A detour through a charging station for the city of Boston.

| Network | Boston | | Dublin | |
|---|---|---|---|---|
| Stats | Mean | Std | Mean | Std |
| $k = 1$ | 769 | 777 | 747 | 863 |
| $k = 2$ | 436 | 541 | 501 | 578 |
| $k = 4$ | 316 | 415 | 298 | 465 |

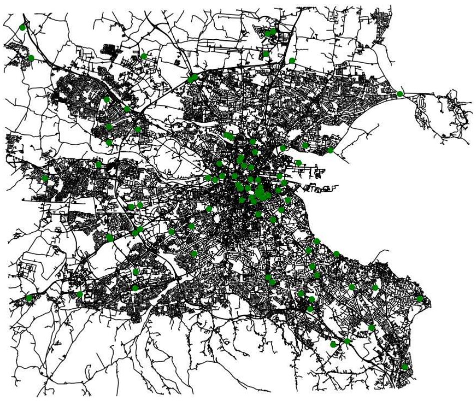
Table 4: Statistics of detours required for recharging batteries for random pairs of source and destination locations in the road networks and different values of $k$.

*5.5. Realistically constrained scenarios*

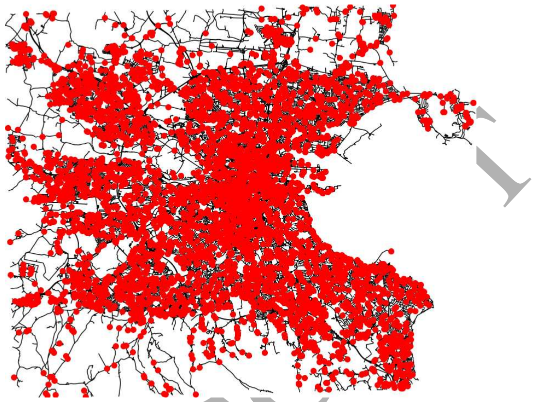
To demonstrate applicability of the proposed general approach, model, and randomized and greedy algorithms, we considered the following constrained real world scenario. We assume there exists a set of already installed charging stations, which we wish to transform into a $k$-dominating set through addition of new charging stations, where the additional charging stations may only be placed at a specified subset of locations. In order to make the proposed randomized and greedy algorithms applicable to this scenario, each of them can be adapted in the following way. First, instead of initializing respectively the sets $A$ and $D$ in the randomized and greedy algorithms to be the empty set, we initialize each of these sets to be the set of already existing charging stations. Second, for adding new charging stations, we only consider a specified subset of locations where additional charging stations may be installed. Finally, when reducing a $k$-dominating set to be minimal, we do not remove any elements belonging to the set of already installed charging stations.

To evaluate the new more constrained model and the adapted randomized and greedy algorithms, we consider the actually installed set of charging stations in the city of Dublin, whose locations can be obtained from the Irish state owned electricity company Electricity Supply Board (ESB). The data in question is freely available from the ESB website in Keyhole Markup Language (KML) format ([13]). The set of installed charging stations has cardinality 90 and is illustrated in Figure 6(a). For this set of charging stations, the minimum number of charging stations within a distance of 3km from any vertex is 0. Therefore, this set of charging stations is not a $k$-dominating set for any value of $k$ with respect to the reachability threshold of 3km considered in this article. We specify the subset of other locations, where additional charging stations may be installed, to be $5,000$ randomly chosen vertices in the road network graph with currently no charging stations installed. Note that, this represents less than 10% of the vertices in the graph, which contains $55,162$ vertices. These locations are illustrated in Figure 6(b).

Table 5 displays the cardinalities of the k-dominating sets computed using the adapted greedy algorithm and the cardinalities of the smallest of ten k-dominating sets computed using the adapted randomized algorithm for each value of $k = 1, 2, 4$. The 2-dominating sets in question are illustrated in Figure 7. By comparing corresponding values in Tables 1 and 5, one can see that the dominating sets are larger in this constrained case scenario than those computed in the unconstrained case. Clearly, this increase is



(a)



(b)

Figure 6: (a) Current set of charging stations installed in the city of Dublin [13] represented by green dots, and (b) $5,000$ locations where new charging stations are allowed for installation represented by red dots.

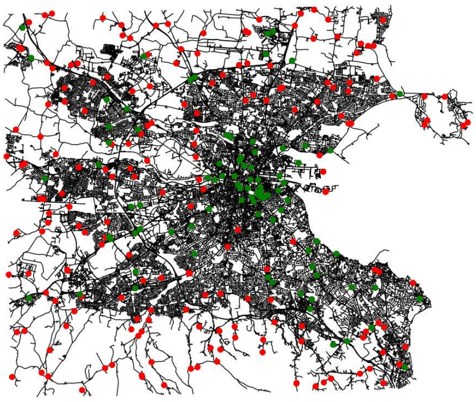
| Network | Dublin | |
|---|---|---|
| Algorithm | Greedy | Randomized |
| $k = 1$ | 178 | 177 |
| $k = 2$ | 277 | 271 |
| $k = 4$ | 456 | 460 |

Table 5: Cardinalities of the $k$-dominating sets computed using the adapted greedy algorithm and the smallest $k$-dominating sets computed using the adapted randomized algorithm for each value of $k$.

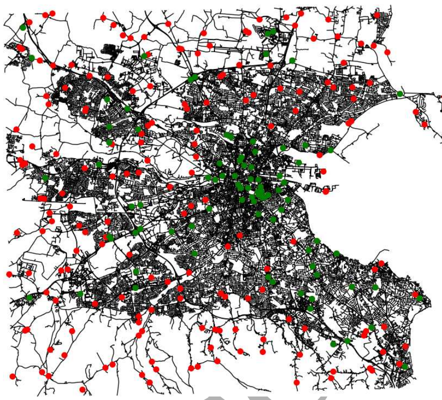
a consequence of the fact that the introduction of constraints reduces the number of possible feasible solutions: the resulting sets must contain the original 90 vertices, and their extension is only possible by using the specified 9.1% of other locations.

## 6. MILP formulations, generalizations, and research perspectives

As an example how possibly to deal with generalizations of the main model of reachability graphs and $k$-dominating sets proposed in this paper, we show how to incorporate the information about demand and capacities into a MILP formulation of the problem. We assume the demand $d_i \geq 0$ and potential capacity $c_i \geq 0$ for installing a charging station at each vertex $v_i$



(a)



(b)

Figure 7: (a) The extended 2-dominating set of 277 vertices computed by the greedy algorithm, and (b) the smallest extended 2-dominating set of 271 vertices computed by the randomized algorithm for the city of Dublin (see Table 5). Green and red dots represent currently installed and new charging stations, respectively.

of the reachability graph $G_t^r$ of the road network is known, $i = 1, 2, \ldots, n$, $|V(G_t^r)| = n$. Suppose a decision maker wants to minimize the total capacity of actually installed charging stations for the road network while preserving the property of $k$-dominating sets in the reachability graph $G_t^r$ and satisfying the demand for charging in the network. We use pseudo-Boolean (0-1) decision variables $x_i$ to indicate whether the charging station of given capacity $c_i$ is installed at vertex $v_i$ of $G_t^r$, $i = 1, 2, \ldots, n$, and real decision variables $y_{ij}$ to indicate proportion ($\times 100\%$ percentage) of the demand $d_i$ at vertex $v_i$ that is supposed to be satisfied by a charging station at vertex $v_j$ of capacity $c_j$, $i, j = 1, 2, \ldots, n$.

Adapting the MILP formulation of the classic transportation problem (e.g., see [7]), the MILP model for this optimization problem can be written as follows:

$$\sum_{i=1}^{n} c_i x_i \longrightarrow \min, \tag{3}$$

subject to

$$\sum_{v_i \in N(v_j)} x_i \geq k(1 - x_j), \quad j = 1, 2, \ldots, n \tag{4}$$

$$\sum_{v_i \in N[v_j]} d_i y_{ij} \leq c_j x_j, \quad j = 1, 2, \ldots, n \tag{5}$$

$$\sum_{v_j \in N[v_i]} y_{ij} = 1, \quad i = 1, 2, \ldots, n \tag{6}$$

$$y_{ij} = 0, \quad v_i v_j \notin E(G_t^r) \;\; i, j = 1, 2, \ldots, n \tag{7}$$

$$y_{ii} \geq x_i, \quad i = 1, 2, \ldots, n \tag{8}$$

$$x_i \in \{0, 1\}, \quad i = 1, 2, \ldots, n \tag{9}$$

$$y_{ij} \in [0, 1], \quad i, j = 1, 2, \ldots, n, \tag{10}$$

where

- (3) is the objective function to minimize the total capacity of installed charging stations in the network to satisfy the demand and to have the $k$-domination property for the set of chosen locations for the whole network (reachability graph $G_t^r$);
- constraints (4) guarantee that either a vertex $v_j$ is in the $k$-dominating set ($x_j = 1$), or else ($x_j = 0$) the vertex $v_j$ is dominated by at least $k$ neighbours in the $k$-dominating set;
- (5) represents the requirement that the total demand satisfied at vertex $v_j$ does not exceed the capacity $c_j$ of $v_j$ (in case a charging station is installed at $v_j$, i.e. when $x_j = 1$);
- constraints (6), (7), and (8) guarantee that the whole demand $d_i$ at vertex $v_i$ can be satisfied at some charging stations installed in the vertices $v_j$ of its closed neighbourhood $N[v_i]$, and if a charging station is installed at $v_i$, the whole demand $d_i$ must be satisfied locally at $v_i$.

If one would like to guarantee that the whole demand of the neighbourhood of a charging station installed at vertex $v_j$ can be eventually satisfied (i.e., in case all the drivers from the reachability distance of $v_j$ decide to go to charge at $v_j$), additional constraints $\sum_{v_i \in N[v_j]} d_i(1 - x_i) \leq c_j$ need to be added to the MILP problem (3)–(10), $j = 1, 2, \ldots, n$. Notice that satisfaction of the total demand in the network, i.e. $\sum_{j=1}^{n} d_j \leq \sum_{j=1}^{n} c_j x_j$, is guaranteed by the constraints (5) taking into consideration (6). We plan to investigate behaviour of this and similar models in future research.

In view of the MILP problem formulation above, the smallest cardinality $k$-dominating set problem in reachability graphs can be formulated as a particular case of (3)–(10) by putting $c_i = 1$ in the objective function (3), and considering only constraints (4) and decision variables $x_i \in \{0, 1\}$ in (9), $i = 1, 2, \ldots, n$. This provides a pseudo-Boolean (0-1) integer programming formulation of the problem considered in the previous sections, which can be used to find exact solutions for the smallest size $k$-dominating sets in graphs.

To be sure the greedy solutions used as a benchmark in the experimental evaluation in Section 5 are meaningful, we have compared greedy solutions with exact solutions for reachability graphs $G_t^r$ corresponding to small-size areas of Boston, using reasonably chosen values for the threshold parameter $t$. The largest size reachability graphs, for which it has been possible to obtain exact solutions for $k = 1, 2$, and 4, correspond to a road network area of about 0.5km by 0.5km with the threshold $t$ equal to 175 and 200 meters. These reachability graphs contain 50 vertices. For these graph instances, the greedy solution either coincides with an exact solution or is at most 3 vertices larger, which is within 8.9% of the optimal exact solutions. For appropriate larger reachability graph instances (up to 146 vertices, corresponding to an area of about 1km by 1km with the threshold of 300 meters), it has been possible to compute exact solutions only for 1-dominating sets, i.e. when $k = 1$: all the corresponding greedy solutions turned out to be within 22.3% of the optimal solutions. We believe these experimental results with exact solutions justify well the choice of greedy heuristic solutions as a benchmark for the large-scale optimization considered in Section 5.

## 7. Conclusions

In this paper, we show analysis and good suitability of multiple domination models for decision problems related to efficient and effective placement of charging stations for electrical vehicles in the road networks, which can be used in the case of new alternative fuel vehicles in general. These results can serve as a first approximation to more complicated mathematical models with real road networks and their constraints. We plan to develop this research in the direction of more subtle road and transportation network models, for example, using digraphs and $\alpha$-domination models.

As the experiments in Section 5 show, the approach becomes computationally more challenging in the case of road network graphs corresponding to larger regions and long distance travels. In this case, to reduce size of the graphs, one can use scaling of the road network, where only some of the road intersections, end points, or, in general, possible locations for charging stations are used as vertices to represent the road network and corresponding reachability graphs. Respectively, in this case, the model can be adapted for standard and slow charging (Levels 1 and 2), where a driver also needs to plan in advance charging of his or her vehicle for several hours.

The experimental results with the road networks of Dublin and Boston indicate that sensitivity of the upper bound of Theorem 1, which is strong in general graphs, can be improved in particular cases. Therefore, we conjecture that more sensitive upper bounds similar to Theorem 1 can be obtained by considering the degree sequence of a graph and some other of its parameters and properties. In particular, it would be interesting to obtain a stronger version of Theorem 1 in the case of reachability graphs corresponding to planar or “almost” planar graphs derived from the spacial layouts of road networks.

Some of the limitations of the general model and experiments presented in this paper are not incorporating capacities and different types of the charging stations, as well as demand for charging, which are left for future more subtle research with this kind of models. Notice that demand for charging is a very complicated stochastic process, depending on many real life situations and scenarios (e.g., big sport events or concerts, road closures, day and night time variations, etc.). Therefore, estimating, forecasting, and modelling demand in an appropriate way is a marvellous problem in itself. For example, see [3, 8, 10, 14]. Our results in Section 6 describe how to generalize the proposed model to incorporate demand, capacities, and possibly some other features by using a classic MILP modelling approach. We plan to investigate behaviour of this and similar models in future research.

**Acknowledgement**
The authors are grateful to the anonymous reviewers for their useful comments and suggestions. Both authors contributed equally to this work.

## References


[1] N. Alon, J.H. Spencer, The Probabilistic Method, John Wiley & Sons Inc., New York, 1992.

[2] N. Alon, Transversal numbers of uniform hypergraphs, Graphs Combin. **6** (1990), 1–4.

[3] J. Asamer, M. Reinthaler, M. Ruthmair, M. Straub, J. Puchinger, Optimizing charging station locations for urban taxi providers, Transportation Research Part A **85** (2016), 233–246.

[4] H. Bast, D. Delling, A. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner, R. Werneck, Route planning in transportation networks, Algorithm Engineering: Selected Results and Surveys, Lecture Notes in Computer Science **9220**, (2016), 19–80.

[5] K. Belous, First electrobuses in the streets of Minsk, "Minsk-News" Agency, May 16, 2017 (Russian), `http://minsknews.by/blog/2017/05/16/page/2/` (accessed Sept 25, 2017)

[6] G. Brandstätter, C. Gambella, M. Leitner, E. Malaguti, F. Masini, J. Puchinger, M. Ruthmair, D. Vigo, Overview of optimization problems in electric car-sharing system design and management, in: H. Dawid et al. (eds.), Dynamic Perspectives on Managerial Decision Making, Dynamic Modeling and Econometrics in Economics and Finance **22**, Springer, 2016, 441–471.

[7] K. Bulbul, G. Ulusoy, A. Sen, Classic transportation problems, in: G. Don Taylor (ed.), Logistics Engineering Handbook, CRC Press, 2008, 16:1–32.

[8] J. Cavadas, G.H. Correia, J. Gouveia, A MIP model for locating slow-charging stations for electric vehicles in urban areas accounting for driver tours, Transportation Research Part E **75** (2015), 188–201.

[9] Chariot e-Bus product description, Chariot Motors Group, `http://www.chariot-electricbus.com/products/chariot-e-bus/` (accessed Sept 25, 2017)

[10] T.D. Chen, K.M. Kockelman, M. Khan, Locating electric vehicle charging stations: parking-based assignment method for Seattle, Washington, Transportation Research Record, No. 2385, (2013), 28–36.

[11] P. Corcoran, P. Mooney, M. Bertolotto, Analysing the growth of OpenStreetMap networks, Spatial Statistics **3** (2013), 21–32.

[12] J.E. Dunbar, D.G. Hoffman, R.C. Laskar, L.R. Markus, $\alpha$-Domination, Discrete Math. **211** (2000), 11–26.

[13] Electricity Supply Board (ESB) web-site, `https://www.esb.ie/electric-cars/kml/charging-locations.kml` (accessed Sept 25, 2017)

[14] I. Frade, A. Ribeiro, G. Gonçalves, A.P. Antunes, Optimal location of charging stations for electric vehicles in a neighborhood in Lisbon, Portugal, Transportation Research Record, No. 2252, (2011), 91–98.

[15] S. Funke, A. Nusser, S. Storandt, Placement of Loading Stations for Electric Vehicles: No Detours Necessary!, Journal of Artificial Intelligence Research **53** (2015), 633–658.

[16] A. Gagarin, A. Poghosyan, V.E. Zverovich, Upper bounds for $\alpha$-domination parameters, Graphs Combin. **25** (2009), 513–520.

[17] A. Gagarin, A. Poghosyan, V. Zverovich, Randomized algorithms and upper bounds for multiple domination in graphs and networks, Discrete Appl. Math. **161** (2013), 604–611.

[18] T. Hamilton, Next stop: ultracapacitor buses, MIT Technology Review, Oct 19, 2009, `https://www.technologyreview.com/s/415773/next-stop-ultracapacitor-buses/` (accessed Sept 25, 2017)

[19] J. Harant, M.A. Henning, A realization algorithm for double domination in graphs, Utilitas Math. **76** (2008), 11–24.

[20] L. He, H.-Y. Mak, Y. Rong, Z.-J.M. Shen, Service region design for urban electric vehicle sharing systems, Manufacturing & Service Operations Management **19**(2) (2017), 309–327.

[21] M. Hosseini, S.A. MirHassani, Refueling-station location problem under uncertainty, Transportation Research Part E **84** (2015), 101–116.

[22] A.Y.S. Lam, Y.-W. Leung, X. Chu, Electric vehicle charging station placement: formulation, complexity, and solutions, IEEE Transactions on Smart Grid **5** (2014), 2846–2856.

[23] J.K. Lan, G.J. Chang, Algorithmic aspects of the $k$-domination problem in graphs, Discrete Appl. Math. **161** (2013), 1513–1520.

[24] X. Li, J. Ma, J. Cui, A. Ghiasi, F. Zhou, Design framework of large-scale one-way electric vehicle sharing systems: a continuum approximation model, Transportation Research Part B **88** (2016), 21–45.

[25] M.K. Lim, H.-Y. Mak, Y. Rong, Toward mass adoption of electric vehicles: impact of the range and resale anxieties, Manufacturing & Service Operations Management **17**(1) (2015), 101–119.

[26] H.-Y. Mak, Y. Rong, Z.-J.M. Shen, Infrastructure planning for electric vehicles with battery swapping, Management Science **59**(7) (2013), 1557–1575.

[27] Y.(M.) Nie, M. Ghamami, A corridor-centric approach to planning electric vehicle charging infrastructure, Transportation Research Part B **57** (2013), 172–190.

[28] A. Poghosyan, D.V. Greetham, S. Haben, T. Lee, Long term individual load forecast under different electrical vehicles uptake scenarios, Applied Energy **157** (2015), 699–709.

[29] S. Storandt, S. Funke, Cruising with a Battery-Powered Vehicle and Not Getting Stranded, In Proc. of the 26*th* AAAI Conference on Artificial Intelligence, Vol. 3. (2012), 1628–1634.

[30] J. Zhao, T. Ma, Optimizing layouts of initial AFV refueling stations targeting different drivers, and experiments with agent-based simulations, European Journal of Operational Research **249**(2) (2016), 706–716.

[31] Zero Emission Urban Bus System (ZeEUS) project, ZeEUS eBus Report #1 (2016), `http://zeeus.eu/uploads/publications/documents/zeeus-ebus-report-internet.pdf` (accessed Sept 25, 2017)